

Archiving Software in PDS

Response to RFI: Preparation for the Development of a Community-Based Roadmap for NASA's Planetary Data Services

Solicitation Number: NNH15ZDA012L

Submitted By: Robert Deen

Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Dr, MS 168-300, Pasadena, CA 91109

Bob.Deen@jpl.nasa.gov

818-354-7492

Summary

The PDS needs to consider how to archive software that processes its data holdings, in addition to the data holdings themselves.

Rationale

As technology moves forward, instruments on planetary missions are continuously increasing in complexity. This complexity is reflected in the data produced by the instruments, which becomes harder and harder to interpret. PDS has the responsibility to archive this data for future generations. As the data becomes more complex, it also becomes increasingly dependent on the software used to process it.

Software can be very complex and expensive to reproduce. Gone are the days when one could take a paper describing an algorithm and implement it in a few days. It could take weeks or months to implement an algorithm from a paper – and most papers provide only an overview of the methodology. For example, the “mars*” programs used by the author’s organization at JPL (MIPL/OPGS – Multimission Image Processing Lab/Operational Product Generation Subsystem) that creates most of the Reduced Data Record (RDR) images in PDS for the Mars surface missions, represent many man-years worth of effort. It would be a significantly non-trivial effort to re-create even a portion of that system.

Even if one were to re-create a portion of the system based on published algorithms, it would be impossible to reproduce them exactly. This calls into question the scientific repeatability of results. If results cannot be duplicated, are the results derived from them scientifically valid? Of course the data products are validated, but even with validation, bugs can occur.

Usability of the data is also a concern. Image data is typically easy to view using any number of tools, but non-image data is significantly less so. Even in the imaging realm, non-traditional images such as XYZ or arm reachability maps are hard to visualize without special tools. The operations and science teams of missions invest significant resources into creating tools that enable viewing and interpretation of data from the mission. By and large, these tools are not available outside the mission team.

There is clearly a need to provide software that accesses, processes, and visualizes PDS data. The 2015 PDART (Planetary Data, Archive, Restoration, and Tools) call from ROSES made this explicit by insisting that new software developed as part of the PDART task be made open source to GitHub. Many software tools are being released in the open source domain even without being forced, via portals like GitHub and SourceForge.

However these repositories are not *archival* in nature. Software “rots” over time if it is not maintained. Operating systems, languages, and libraries evolve, and unmaintained code becomes unusable before long. On the time scale of planetary data archives (multiple decades), odds are the current open source repositories won’t even exist – but even if they did, the planetary code contained in them would no longer compile and run without continual maintenance.

Suggestions

I recommend that PDS undertake a study to determine how software used to access, process, analyze and visualize its data holdings can be archived. It might even be advantageous to create a PDS Software Node to manage software archiving and curation in a consistent way.

In addition to the “bit rot” problem mentioned above, complex software often depends on libraries, and these too must be preserved for the software to be able to compile and run later. Sometimes these libraries have licensing issues – issues that get very murky when the license holder no longer exists (as so often happens with software companies over time), or no longer supports the code.

One potential way to handle the problem is to preserve an entire executable (and compilation) environment as a virtual machine, with the hope that the entire virtual machine could be revived as needed in the future. However, the odds of the format in which the virtual machine is stored being viable decades from now are pretty low.

This is not an easy problem, and I do not have solutions to recommend. However, it is a serious problem, which will continue to get worse as instruments and software get more complex, and PDS needs to start thinking about how to address it.

Impacts

Specialized software is increasingly important for processing the PDS data holdings. If the software is not archived in some way, the tools needed to interpret archived data will become unavailable over the lifetime of the PDS archive. This would be a disservice to future generations, as they would have to expend significant effort to re-create the software that was lost in order to effectively use the data holdings.

About the Author

Robert Deen is a Principal at the JPL Multimission Image Processing Lab, where he’s been since 1987. He is the lead developer for the ground-based image processing software used by all the recent Mars *in-situ* missions (MER, PHX, MSL, InSight, Mars 2020) for operations and science processing. This software does stereo processing, meshes, mosaics, radiometric correction, and higher-order products such as slope maps and arm reachability products. He is also on the operations team for all these missions. He is instrumental in PDS image label design for these missions, and wrote much of the technical content for the SIS (Software Interface Specification) documents describing the imaging data for them. He is the PDS data provider for the MSL PLACES rover localization database. Recently he has been involved in the PDS 4 transition effort for InSight, and occasionally contributes to PDS design discussions.

Acknowledgments

The author recognized the need for archiving software as a result of viewing a poster on the topic by Chase Million at the 2015 Planetary Data Workshop in Flagstaff, AZ.